Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of the Claims:

1. (cancelled)

2. (cancelled)

3. (currently amended) A method of generating an interleave pattern for n lots of A tasks and y lots of B tasks, where n, y, and z are positive integers, B equals A plus 1, and n plus y does not equal a power of two, comprising:

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting ~~The method of claim 2 wherein said selecting includes selecting~~ a centered portion of the list;

renumbering the selected portion of the list to form a key;

generating an interleave pattern corresponding to said key in which all values in the key less than n are replaced by A and all other values in the key are replaced by B; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said interleave pattern to balance the workload across said plurality of processing elements.

4. (original) A method of generating an interleave pattern for n lots of A tasks and y lots of B tasks, where n, y, and z are positive integers, B equals A plus 1, and n plus y does not equal a power of two, comprising:

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting a portion of the list by ~~The method of claim 2 wherein said selecting includes~~ dropping entries alternately from each side of the list;

renumbering the selected portion of the list to form a key;

generating an interleave pattern corresponding to said key in which all values in the key less than n are replaced by A and all other values in the key are replaced by B; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said interleave pattern to balance the workload across said plurality of processing elements.

5. (currently amended) The method of claim [[2]] 3 wherein said renumbering includes renumbering in order of ascending value.

6. – 7. (cancelled)

8. (currently amended) A method, comprising:

creating a key comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

creating a table of interleave patterns for all values of n lots of A tasks and $(2^z - n)$ lots of B tasks based on said key, where n and z are positive integers and B equals A + 1;

storing said table;

automatically selecting an interleave pattern from said table based on one of the values n and $(2^z - n)$;

~~The method of claim 7 additionally comprising~~ generating an interleave pattern based on said selecting; and

distributing said n lots of A tasks and said $(2^z - n)$ lots of B tasks to a plurality of processing elements according to said generated interleave pattern to balance the workload across said plurality of processing elements.

9. (cancelled)

10. (currently amended) A method, comprising:

selecting a value of 2^z which is greater than the value of n lots of A tasks plus y lots of B tasks, where n, y, and z are positive integers and B equals A plus 1, but less than twice that value;

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting ~~The method of claim 9 wherein said selecting includes select~~ing a centered portion of the list;

renumbering the selected portion of the list to form a key;

creating a table of interleave patterns for all values of n lots of A and y lots of B based on said key;

storing said table; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said stored table to balance the workload across said plurality of processing elements.

11. (currently amended) A method, comprising:

selecting a value of 2^z which is greater than the value of n lots of A tasks plus y lots of B tasks, where n, y, and z are positive integers and B equals A plus 1, but less than twice that value;

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting a portion of the list by ~~The method of claim 9 wherein said selecting includes~~ dropping entries alternately from each side of the list;

renumbering the selected portion of the list to form a key;

creating a table of interleave patterns for all values of n lots of A and y lots of B based on said key;

storing said table; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said stored table to balance the workload across said plurality of processing elements.

12. (currently amended) The method of claim [[9]] 10 wherein said renumbering includes renumbering in order of ascending value.

13. (currently amended) A method, comprising:

selecting a value of 2^z which is greater than the value of n lots of A tasks plus y lots of B tasks, where n, y, and z are positive integers and B equals A plus 1, but less than twice that value;

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting a portion of the list;

creating a table of interleave patterns for all values of n lots of A and y lots of B based on said key;

storing said table;

~~The method of claim 9 additionally comprising~~ automatically selecting an interleave pattern from said table based on one of the values n and y;

generating an interleave pattern based on said selecting; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said generated interleaved pattern table to balance the workload across said plurality of processing elements.

14.-15. (cancelled)

16. (new) The method of claim 4 wherein said renumbering includes renumbering in order of ascending value.

17. (new) The method of claim 11 wherein said renumbering includes renumbering in order of ascending value.

18. (new) A computer memory device storing a set of instructions which, when executed, performs a method of generating an interleave pattern for n lots of A tasks and y lots of B tasks, where n, y, and z are positive integers, B equals A plus 1, and n plus y does not equal a power of two, said method comprising:

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting a centered portion of the list;

renumbering the selected portion of the list to form a key;

generating an interleave pattern corresponding to said key in which all values in the key less than n are replaced by A and all other values in the key are replaced by B; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said interleave pattern to balance the workload across said plurality of processing elements.

19. (new) A computer memory device storing a set of instructions which, when executed, performs a method of generating an interleave pattern for n lots of A tasks and y lots of B tasks, where n, y, and z are positive integers, B equals A plus 1, and n plus y does not equal a power of two, said method comprising:

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting a portion of the list by dropping entries alternately from each side of the list;

renumbering the selected portion of the list to form a key;

generating an interleave pattern corresponding to said key in which all values in the key less than n are replaced by A and all other values in the key are replaced by B; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said interleave pattern to balance the workload across said plurality of processing elements.

20. (new) A computer memory device storing a set of instructions which, when executed, performs a method comprising:

creating a key comprised of the reverse bit order of a serially indexed count from 0 to 2^z;:

creating a table of interleave patterns for all values of n lots of A tasks and $(2^z - n)$ lots of

B tasks based on said key, where n and z are positive integers and B equals A + 1;

storing said table;

automatically selecting an interleave pattern from said table based on one of the values n and $(2^z - n)$;

generating an interleave pattern based on said selecting; and

distributing said n lots of A tasks and said $(2^z - n)$ lots of B tasks to a plurality of processing elements according to said generated interleave pattern to balance the workload across said plurality of processing elements.

21. (new) A computer memory device storing a set of instructions which, when executed, performs a method comprising:

selecting a value of 2^z which is greater than the value of n lots of A tasks plus y lots of B tasks, where n, y, and z are positive integers and B equals A plus 1, but less than twice that value;

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting a centered portion of the list;

renumbering the selected portion of the list to form a key;

creating a table of interleave patterns for all values of n lots of A and y lots of B based on said key;

storing said table; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said stored table to balance the workload across said plurality of processing elements.

22. (new) A computer memory device storing a set of instructions which, when executed, performs a method comprising:

selecting a value of 2^z which is greater than the value of n lots of A tasks plus y lots of B tasks, where n, y, and z are positive integers and B equals A plus 1, but less than twice that value;

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z, where z is an integer;

selecting a portion of the list by dropping entries alternately from each side of the list;

renumbering the selected portion of the list to form a key;

creating a table of interleave patterns for all values of n lots of A and y lots of B based on said key;

storing said table; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said stored table to balance the workload across said plurality of processing elements.

23. (new) A computer memory device storing a set of instructions which, when executed, performs a method, comprising:

selecting a value of 2^z which is greater than the value of n lots of A tasks plus y lots of B tasks, where n, y, and z are positive integers and B equals A plus 1, but less than twice that value;

creating a list in which the entries are comprised of the reverse bit order of a serially indexed count from 0 to 2^z;

selecting a portion of the list;

creating a table of interleave patterns for all values of n lots of A and y lots of B based on said key;

storing said table;

automatically selecting an interleave pattern from said table based on one of the values n and y;

generating an interleave pattern based on said selecting; and

distributing said n lots of A tasks and said y lots of B tasks to a plurality of processing elements according to said generated interleaved pattern table to balance the workload across said plurality of processing elements.